

VŠB – Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra informatiky

Absolvování individuální odborné praxe  
Individual Professional Practice in the  
Company

2011

Michal Ševčík

## Prohlášení studenta

Prohlašuji, že jsem tuto bakalářskou/diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

Dne:

.....

Podpis

# Poděkování

Děkuji vedoucí bakalářské práce Ing. Emilii Šeptákové za cennou pomoc při zpracování závěrečné zprávy.

Dále děkuji všem kolegům z divize Vývoj softwaru, kteří mi v průběhu praxe poskytli velmi užitečné informace.

Nakonec bych chtěl poděkovat své rodině a přátelům za pochopení nedostatku času a ohromnou podporu.

# Abstrakt

V této práci autor popisuje průběh jeho odborné praxe. Seznamuje čtenáře se společnostmi, u kterých praxi vykonával, se zadanými úkoly a následně s postupy řešení. Uvádí technologie a postupy, které při práci používal. Na konci zprávy autor shrnuje dosažené výsledky v celkovém zhodnocení.

## Klíčová slova

Odborná praxe, AutoCont, .NET, SQL, ASP .NET, WPF, C#, XAML, *Valeo*, *BukacHockey*, *JagrTeam*

## Abstract

In this dissertation the author describes a process of his practical training experience. He presents the company he worked for. First chapters of the dissertation contain tasks that the author has received and his methods of solution. In following chapters the author specifies the technologies and methodology he has used. In the last part the author summarizes all working results.

## Key Words

Professional practice, AutoCont, .NET, SQL, ASP .NET, WPF, C#, XAML, *Valeo*, *BukacHockey*, *JagrTeam*

# Seznam použitých symbolů a zkratek

(G)UI	(graphical) user interface, grafické uživatelské rozhraní
.NET Framework	zastřešující název pro soubor technologií v softwarových produktech, vyvíjených společností Microsoft
ASP .NET	součást .NET Frameworku sloužící k vývoji webových aplikací a služeb
C#	objektově orientovaný programovací jazyk vyvinutý firmou Microsoft
CLR	Common Language Runtime
DDL	Dynamic-Link Library
Desktop (windows) aplikace	klasické aplikace běžící na klientském počítači
Garbage Collector	automatický nástroj pro správu paměti
JavaScript	klientský programovací jazyk
LightBox	Java scriptová knihovna pro uživatelsky přívětivé zobrazení fotogalerií
LINQ	viz. níže
MS SQL	Microsoft SQL server, databázový server společnosti Microsoft
MVVM	Model-View-ViewModel, viz. níže
OOP	objektově orientované programování
SQL	standardizovaný dotazovací jazyk používaný při práci s relačními databázemi
SW	software, počítačový program
Video Streaming	technologie způsobu přenášení video obsahu od zdroje k uživateli v prostředí internetu
WPF	podmnožina .NET Frameworku používající XAML pro uživatelsky bohaté rozhraní
WYSIWYG	akronym věty „What you see is what you get“, zkratka označuje skupinu editorů, které editovaný obsah zobrazují ve finální formě
XAML	značkovací jazyk vyvíjený firmou Microsoft založený na XML
XML	standardizovaný obecný značkovací jazyk

# Obsah

1.	Úvod .....	1
2.	Zadané úkoly a jejich řešení .....	3
2.1.	Projekt Valeo .....	3
2.2.	Projekt BukačHockey .....	5
2.3.	Projekt JágrTeam .....	6
3.	Použité technologie I .....	9
3.1.	C# .....	9
3.2.	ASP.NET .....	10
3.3.	T-SQL .....	11
4.	Použité technologie II .....	12
4.1.	MVVM .....	12
4.2.	LINQ .....	13
5.	Závěr .....	14
6.	Použitá literatura .....	14

# 1. Úvod

Absolvování odborné praxe ve společnosti je alternativa k vypracování teoretické práce na zadané téma, vůči které má mnoho výhod pro studenta samotného, který takto získá potřebné informace a zkušenosti z oboru v ostrém firemním provozu.

Nastoupil jsem do společnosti AutoCont CZ a.s. Tato se zaměřuje na služby a řešení v oblasti informačních technologií. Společnost je rozčleněna na několik samostatných divizí s vlastní samosprávou. Těmi jsou:

- **IT infrastruktura** – čítá dohromady 350 zaměstnanců a je proto největší divizí společnosti. Tato divize poskytuje specializované a profesionální služby spojené s celým životním cyklem IT infrastruktury. Od dodávky samotného IT zařízení, přes návrh a realizaci infrastrukturních řešení až po správu a technickou podporu výsledného provozu. Výsledkem činnosti divize je pak poskytování stabilního, bezpečného a uživatelsky přívětivého prostředí na serverech, klientských stanicích i mobilních zařízeních za přiměřené náklady. Toto všechno může divize poskytovat především díky rozsáhlé obchodní síti, jež se skládá z 8 regionálních center a z 8 specializovaných odborných center.
- **Outsourcing** – tato divize se zaměřuje na dodávku služeb v oblasti provozu informačních systémů u velkých zákazníků. Těžiště nabízených služeb spočívá v provádění správy a údržby informačního systému zákazníka nebo jeho části. Součástí nabízených služeb je také forma kompletního outsourcingu provozu informačního systému včetně převzetí majetku a pracovníků.
- **Divize podnikové aplikace a služby** - hlavní činností divize PAS je implementace, podpora a rozvoj podnikových informačních systémů. Jedná se o vzájemně propojené informační systémy, složené z aplikací pro ekonomiku, plánování a řízení zdrojů (ERP), z personálního informačního systému, dále informačních systémů pro kolektivní spolupráci, organizaci práce a komunikaci a manažerských informačních systémů. Jádrem takto vzniklého informačního systému je MS Dynamics AX nebo Microsoft Dynamics NAV doplněný o řešení na bázi aplikací MS SharePoint, MS Project a Project Server a řady nástrojů pro Business Intelligence – MS SQL Server R2, MS Office 2010 a MS Share Point 2010. Součástí může nově být personální systém Magma HCM, jehož výrobcem je sesterská společnost versity.
- **Content Management** – divize zabývající se poskytováním kvalitních a komplexních služeb pro korporace a organizace státní nebo veřejné správy v oblasti uchovávání, ochrany, poskytování a uložení dat v prostředí společnosti
- **SAP** – se zabývá řešením problematiky spojené s touto platformou, která je bohatě využívána v oblasti středních až velkých firem.
- **Vývoj softwaru** - se zabývá poskytováním komplexních služeb v oblasti vývoje softwaru na zakázku. Přináší mnoha firmám a organizacím konkurenční výhodu prostřednictvím řešení stavěných na míru pro konkrétní klienty, řešení s vysokou přidanou hodnotou, kterou nemohou získat nákupem standardních produktů. Do této divize jsem nastoupil svou praxi na pozici Junior Vývojář.
- **Centrální prodej** - primárně se zaměřuje na malé organizace, které často nemají své vlastní IT specialisty. Pro podnikatele, malé firmy, města, obce, školy a další vzdělávací zařízení nabízí rychlé dodávky software, hardware a moderní IT služby.

V první části práce popisuji úkoly, které mi byly postupně předkládány k vypracování. Následně ke každému uvádím svůj postup a případné problémy v průběhu řešení.

Ve třetí kapitole vypisuji jednotlivé technologie, které jsem využíval a o kterých jsem se zároveň dozvídal z výuky ve škole.

Čtvrtou kapitolu vyhrazuji technologiím a postupům, které pro mne byly v době nastoupení praxe neznámé, ale pro splnění daných úkolů nepostradatelné.

V závěru jsem shrnul své výsledky a sepsal hodnocení praxe z mé strany.



## 2. Zadané úkoly a jejich řešení

### 2.1. Projekt Valeo

Od firmy Valeo byl firmě AutoCont CZ zadán úkol vytvořit desktop aplikaci umožňující ovládání celé výrobní linky, tzv. CSK. Samotná linka se skládá z několika samostatných výrobních stanic od různých výrobců. Hlavním úkolem CSK je tedy sjednotit kontrolu a data všech částí linky do jednoho společného bodu, ze kterého se bude moci ovládat celá linka.

Aplikace samotná obsahuje dílčí moduly (dle návrhového vzoru MVVM, viz níže), kdy každý modul je zodpovědný za určitou část linky. Moduly jsou rozčleněny do několika kategorií v závislosti na funkci, kterou zprostředkovávají. Těmito kategoriemi jsou:

- System Settings – centrální nastavení linky;
- Components – užitečné funkce aplikace, např. zobrazení párů čekajících na zpracování, zobrazení virtuálních rotorů nebo export konfigurace do souboru;
- Tracing – umožňuje zobrazit historii jednotlivých vyráběných dílů;
- Setup production – nastavení výrobního plánu;
- CSK – informace o lince;
- Logs – logování běhu linky;
- Partial Store – zobrazení stavu skladů.

Podílel jsem se na vývoji a testování trasovacího modulu, modulu pro správu uživatelů a modulu pro zobrazení chybových zpráv.

#### **Trasovací modul**

Trasovací modul se skládá ze tří funkčně identických částí, každá ale zobrazuje historii jiných výrobních komponent (Basket – košík až 10-ti částí, Cast - odlitek, Frame – rám až 10-ti basketů). Celá aplikace operuje nad MS SQL databází, do které se zapisují všechna data o výrobním procesu, včetně data poslední změny. Trasování jednotlivých komponent tedy není nijak komplikované z této databáze vyčíst a následně ve formě tabulky zobrazit.

Vyhledávacími kritérii jsou: rozmezí data od-do, identifikační čísla komponent, stavy výroby komponenty.

#### **Modul pro správu uživatelů**

Jak již název napovídá, tento modul má na starosti vytváření, editaci a mazání uživatelů schopných se do systému přihlásit. V záložce určené tomuto modulu je v tabulce zobrazen výpis všech registrovaných uživatelů, na levé straně se nachází panel s několika tlačítky pro ovládání modulu, v tomto případě to jsou tlačítka *New*, *Edit* a *Delete*. Funkce prvních dvou je z názvu patrná, tlačítko pro smazání uživatele však fyzický záznam z tabulky nemaže, nýbrž nastaví příznak aktivity uživatele na *False*. Uživatel nemůže být z tabulky fyzicky smazán, protože se na uživatele váží a odkazují všechny činnosti, které za dobu svého působení v programu vykonal. Jeho smazání by tedy mohlo způsobit narušení těchto vazeb mezi daty v databázi, což je jev, kterému se snažíme vyhnout.

Každý uživatel má několik vlastností – login, jméno, příjmení, heslo a roli. Na základě role jsou uživateli přístupné odpovídající oprávnění v systému. Jednotlivé role jsou: *Guest*, *Operator*, *Supervizor*, *Admin*. Je totiž samozřejmé, že uživatel v roli *Guest* či *Operator* by neměli mít přístup ke správě uživatelů nebo nastavení celé linky.

Aby bylo zamezeno případnému úniku hesel z databáze, jsou uložena v podobě hash kódů, což znamená, že jakmile se jednou heslo uloží, již nikde není uchováváno v podobě čistého textu a jeho odtažení tedy touto cestou není možné. Šifrování obstarává uložená procedura na straně MS SQL.

## 2.2. Projekt BukačHockey

BukačHockey je webová prezentace hokejové školy stejného jména postavená na technologii ASP.NET. Prezentace se skládá z devíti kategorií, přičemž některé jsou ještě dále děleny na podkategorie (*Trénink – na ledě / na suchu*).

Prezentace je přístupná ve třech jazykových verzích – v češtině, angličtině a němčině. Jazyk se volí na úvodní stránce. Aby na serveru nemusely být uloženy tři verze každé stránky, je volba jazyka řešena technikou tzv. globalizace, kdy jsou samotné texty uloženy v tzv. *Resources* aplikace, ze které se dynamicky načítají podle aktuálního jazyka aktuální stránky. Díky tomu tedy máme pro každou jazykovou verzi stejnou stránku, jen se podle příznaku jazyka přistupuje k jinému *Resource*.

Mým úkolem na tomto projektu byla úprava několika textů, úplné přepracování sekce *Média – Video* a přidání a naformátování obsahu do sekce *Odkazy*.

### Úprava textů

Zákazníkem dodanými texty jsem nahradil stávající v již zmíněných *Resources*. Tyto jsem však musel ještě převést do jazyka HTML. Zákazník totiž texty vytvářel kancelářským textovým nástrojem, jehož formátování je nepoužitelné pro webovou prezentaci.

### Přepracování sekce *Media – Video*

Původně tato sekce obsahovala videa uložená na serveru ve formátu .avi a přehrávala je díky plug-inu Media Playeru v prohlížeči. Tento způsob přehrávání videa ale již neplní požadavky dnešního návštěvníka webových stránek a zároveň je obtížnější na správu (v případě většího množství videí či jejich větší délky není jisté, zda webový hosting bude stačit kapacitou). Proto jsem tuto sekci přepracoval tak, aby obsahovala vložená videa ze serveru YouTube, který poskytuje streamovaná videa.

Využití streamovaného obsahu je výhodnější hned z několika důvodů. Asi největší výhodou je fakt, že nenutí uživatele nic fyzicky ukládat na disk a následně video přehrávat ve vlastním přehrávači. Navíc je také možné, že uživatel nemůže na koncovou stanici ukládat žádná stažená data nebo stanice není vybavena video přehrávačem (internetové kavárny). Streamované video se totiž přehrává přímo z webové stránky v prohlížeči, je potřeba mít jen nainstalovanou technologii Adobe Flash, která je ale dnes již nedílnou součástí základní výbavy počítače připojeného k internetu. Jako největší nevýhodu bych označil jisté riziko výpadku serveru třetí strany a tedy nedostupnost videa. Toto riziko je však při výběru nejznámějších serverů (YouTube, Vimeo, ...) díky redundantnosti dat takřka nulové (v případě výpadku serveru jeho roli zaujímá jiný se stejnými daty).

## 2.3. Projekt JágTeam

JágTeam je již existující webová aplikace, kterou si zákazník přál rozšířit o novou funkčnost a také upravit jeho design. Stávající řešení se skládá z několika částí: E-Shop, VIP-Shop, Hokejová škola, Administrace a nakonec je zde i sekce určena biografii Jaromíra Jágra. Novinkou, kterou si zákazník žádá, je Klubový servis. Celá aplikace je postavena na technologii ASP.NET s datovou základnou v relační databázi MS SQL, se kterou komunikuje metodou LINQ.

### Biografie Jaromíra Jágra

V této sekci je momentálně ve stručnosti shrnuta Jágrova kariéra, doplněna o pár fotek. Stávající rozvržení stránky je graficky odlišné od zbytku portálu. Rozdělením na několik podkategorií a sladěním designu s hlavní stránkou portálu tak zákazník chce udělat tuto sekci nejen přehlednější, ale také zajímavější, a to jak z pohledu obsáhlosti informací, tak logiky uspořádání.

Nově tedy přibude členění na kategorie. První z nich bude *Fotogalerie*, kde, jak již název napovídá, budou umístěny různé fotografie. Sekce s názvem *Video* bude obsahovat různá zajímavá videa. V kategorii *Kluby* budou přehledně zobrazeny informace o jednotlivých klubech, ve kterých Jaromír Jág hrál. Jak se dařilo jemu, týmu, apod. Poslední z dílčích sekcí bude *Životopis*, kde bude text zachycující Jágrův život.

### E-Shop

Klasický e-shop s několika funkcemi navíc zaměřený na široké spektrum zákazníků. Nákup je umožněn i neregistrovaným zákazníkům, registrovaní však mají několik výhod.

Zde bylo mým prvním úkolem rozšířit detaily zobrazované ke každému produktu o další kartu, ve které se budou zobrazovat videa, pokud budou přiložena.

Druhým úkolem bylo upravit fotografie do čtyř různých velikostí a přejmenovat je podle názvů produktů. Vzhledem k velkému množství produktů a tedy i fotografií byl tento úkol velmi časově náročný. Po úpravě velikostí fotek je bylo nutné sjednotit do odpovídajících adresářů právě podle rozměrů. Každý adresář je určen pro jinou část e-shopu. Do základního dlaždicového zobrazení produktů v kategorii je použit rozměr 140x140, v náhledu na produkt je umístěn větší obrázek, tzv. *Preview* (280x280) a řada menších náhledů, tzv. *Thumbnails* (70x70), které se po kliknutí na ně přemístí do *Preview*. Posledním a největším rozměrem je 640x640 v tzv. *Full*, tento se zobrazí po kliknutí na *Preview* v tzv. *LightBoxu*.

## VIP-Shop

Je obdobou klasického e-shopu s tím rozdílem, že do toho obchodu mají přístup pouze velkoodběratelé. Můžeme tedy říci, že VIP-Shop plní roli velkoobchodu.

Stávající řešení je zcela samostatná aplikace, kterou jsem dostal za úkol zaintegrovat do společné databáze a upravit tak, aby design byl shodný s E-Shopem. Funkčně se VIP-Shop příliš neliší, rozdíly jsou především v možnostech zákazníka a cenách.

### *Vybrané možnosti zákazníka*

Faktury po splatnosti – upozorňování zákazníka na nesplacené faktury. Faktury importované ze systému třetí strany.

Archiv – různé materiály ke stažení

## Hokejová škola

Tato sekce se zabývá hokejovou školou Jaromíra Jágra. V této sekci najdeme ceník, termíny kurzů, způsob tréninků, doprovodný program, přihlášku na kurz a samozřejmě také kontakty. Zde bylo mým úkolem rozšířit stávající řešení o sekci *Video* a *Fotogalerie*.

## Administrace

Administrační rozhraní, skrz které se celý portál řídí. Hlavním navigačním prvkem je zde tradiční menu na levé straně, které nám také administraci opticky rozděluje do logických skupin (E-shop, VIP-shop, atd.).

### *E-Shop*

V této skupině se nachází export zboží ze systému třetí strany – uživatel vybere zvláštní soubor s pevně daným formátováním, ze kterého se naimportují data do databáze; správa balíčků – správce obchodu se může rozhodnout nabízet určitý balíček zboží výhodněji při společné koupi, tehdy přistoupí k vytvoření balíčku, nastavení parametrů a konečně vystavení na e-shopu; správa produktových řad – umožňuje spravovat jednotlivé kusy zboží, nahrávat obrázky a upravovat cenu či popisy zobrazené při náhledu na zboží; správa uživatelů – kompletní přehled všech registrovaných uživatelů, možnost jakékoliv editace informací, také se zde zobrazuje obrat ke každému zákazníkovi, který je využit při výpočtu slevy; přehled objednávek – zde je v seznamu zobrazen přehled veškerých objednávek vedených v systému, v podrobnostech objednávky je zobrazen objednavatel, objednané zboží, celková cena a další nezbytné informace k bezchybnému provozu.

Ačkoliv je editovatelnost veškerých seznamů kompletní, mazání zde není z důvodu existence pevných vazeb (zákazník – objednávka; produkt – balíček) možné. Bylo tedy nutno zavést v databázi tzv. příznak smazání, na základě kterého jsou vybírány aktivní, neboli nesmazané položky, které se zobrazí v seznamech. V případě, že je tento příznak smazání nastaven na hodnotu PRAVDA(TRUE), objekt(uživatel/produkt/objednávka) se v seznamu nezobrazí, ale systém zůstává konzistentní.

### *VIP-Shop*

Funkcionalita prakticky ekvivalentní k E-shopu, jen se některé operace provádějí nad jinými daty.

### *Hokejová škola*

V této skupině je možno editovat termíny a ceny jednotlivých kurzů hokejové školy. Také je zde administrace fotogalerie a videogalerie.

Do každé výše zmíněné skupiny jsem měl také zavést možnost editovatelnosti veškerých statických textů, které do dané skupiny patřily. Aby toto bylo možné, bylo nutné upravit databázi, do které se texty budou ukládat a následně se z ní při načítání obsahu natahovat do zdrojového kódu stránky. V každé sekci jsem proto umístil odkaz *Statické stránky*, po jehož rozkliknutí se u horního okraje hlavní části webové stránky objeví skupina tzv. radio buttonu (přepínače), na základě kterých se načte text z databáze do níže umístěného online WYSIWYG editoru. Tento prvek je tvoren JavaScriptem a umožňuje kompletní editaci textu podobně jako klasická kancelářská aplikace. Je zde možno formátovat text do odstavců, formovat nadpisy, vkládat obrázky či hypertextové odkazy. Velkou výhodou je pak velmi snadná úprava editovaného textu z textové formy do HTML jazyka, tak, aby se text zobrazil stejně také na webových stránkách.

## **Klubový servis**

Novinka pro hokejové kluby spolupracující s firmou JágrTeam. Výsledná aplikace by měla zastat komplexní nástroj pro správu hráčské výstroje s možností on-line nákupu a reklamace. Druhořadým cílem je poskytnutí přehled o nákupech a efektivitě reklamací manažerům. Klíčovým datovým výstupem z této agendy jsou správně strukturované manažerské reporty o spotřebě a struktuře spotřeby výstroje v daném klubu v dané sezóně.

Realizace plánována až po skončení mé praxe.

## 3. Použité technologie I

### 3.1. C#

*„Je vysokoúrovňový programovací jazyk založený na jazycích C++ a Java, proto je nepřímým potomkem jazyka C, ze které čerpá syntaxi.“* Jazyk najde své využití v prakticky všech IT oblastech. Je možno jej využít nejen pro tvorbu nejběžnějších desktopových aplikací, ale své uplatnění ukáže také na pozadí webových projektů či programů založených na komunikaci s relačními databázemi. V dnešní době velkého rozmachu tzv. smartphones jej lze také uplatnit při vývoji aplikací pro tyto chytré mobilní přístroje.

*Cíle jazyka:*

- jednoduchý, moderní, víceúčelový a objektově orientovaný programovací jazyk
- robustnost, trvanlivost a programátorská produktivita
- jazyk poskytuje podporu pro principy softwarového inženýrství, jako jsou: hlídání hranic polí, detekce neinicializovaných proměnných a automatický garbage collector

*Vlastnosti jazyka*

- neexistuje vícenásobná dědičnost
- třída může implementovat libovolný počet rozhraní
- v OOP se z důvodu zapouzdření často používají tzv. settery a gettery, vnitřní metody, skrz které je jediná přístupová cesta k samotným proměnným. C# zavedl tzv. Property, které se navenek tváří jako proměnné, ale uvnitř těchto properties si můžeme nadefinovat vlastní get a set metody, ve kterých si můžeme značně usnadnit práci s proměnnými
- c# je typově bezpečný, nedovoluje potenciálně nebezpečné implicitní konverze mezi datovými typy
- nepotřebuje dopřednou deklaraci proměnných a metod

## 3.2. ASP.NET

ASP.NET je součástí .NET Frameworku zaměřená na tvorbu webových aplikací. Nahradila starší technologii ASP.

### *Principy*

- technologie založena na CLR, projekty nad touto technologií tedy mohou být psány v jakémkoliv jazyce CLR podporující (Visual Basic, C#, mutace Pythonu, ...);
- aplikace jsou předkompilovávány do několika málo DLL souborů a jsou proto rychlejší než aplikace postavené na skriptovacích jazycích;
- koncept ASP.NET WebForms připodobňuje programátorům práci s webovou stránkou k práci s klasickými windows aplikacemi, zavádí tzv. *Controls*, se kterými programátor pracuje velmi podobně jako v případě oné klasické desktopové aplikace. Při kompilaci se pak vše převede na korektní HTML prvky, přičemž funkčnost zůstává samozřejmě zachována.

### *Výhody*

- rychlost vykonávání kódu,
- kompilace také odhalí mnoho chyb před samotným spuštěním aplikace,
- možnost vytvářet šablony často používaných prvků,
- velká podobnost s programováním klasických Windows aplikací usnadňuje přechod,
- bohatý výběr knihoven tříd a ovládacích prvků,
- široká podpora programovacích jazyků,
- vestavěná podpora dočasné paměti, tzv. *Cache*,
- multiplatformnost,
- od verze 2 generuje validní kód stránek.



### 3.3. T-SQL

Neboli Transact-SQL je proprietární rozšíření do SQL od Microsoftu. Rozšiřuje SQL o procedurální logiku, proměnné, rozličnou podporu pro práci s textovými řetězci, zpracovávání dat, podporu matematických operací, atd. Také upravuje DELETE a UPDATE příkazy. Všechny tyto úpravy činí T-SQL Turing-kompletní.

T-SQL se používá na MS SQL Serveru, na který všechny aplikace se serverem komunikující zasílají T-SQL příkazy bez ohledu na UI aplikace.

#### *Řízení toku dat*

- příkazy BEGIN a END pro označení začátku a konce bloku kódu,
- příkazy IF a ELSE pro větvení programu,
- příkaz WHILE pro cyklení uvnitř procedury,
- příkaz RETURN pro návrat z procedury,
- a další...

#### *Změny v DELETE a UPDATE příkazech*

- T-SQL umožňuje přidání SELECT klauzule, čímž podporuje spojování tabulek.

#### *Bulk Insert*

- T-SQL příkaz pro hromadné vkládání dat do databáze či čtení dat z externího sekvenčního souboru;
- ulehčuje datovému toku od klienta do databáze (namísto i několika tisíc INSERT příkazů se pošle jen jeden BULK INSERT).

## 4. Použité technologie II

### 4.1. MVVM

Model-View-ViewModel je návrhový vzor pro WPF aplikace, jenž je využitelný i pro aplikace postavené na technologii Silverlight. Tento vzor nabízí řešení, jak oddělit logiku aplikace od uživatelského rozhraní a tím psát kód přehlednější jak pro autory, tak pro programátory, kteří ke kódu přijdou až po jeho napsání. MVVM dělí aplikaci do tří vrstev: *Model*, *View*, *ViewModel* (viz. níže).

Samotné WPF je pro tento návrhový vzor přizpůsobeno, proto ve View můžeme jednotlivým UI prvkům přiřadit *Binding* – přiřazení proměnné z ViewModelu, ve které jsou data pro tento prvek a *Command* – čímž řekneme UI prvku, jakou metodu z ViewModelu zavolat při určité události (např. stisknutí tlačítka).

**Model** obstarává referenci na zdroj dat. Zde se řeší fyzický přístup na relační databázi, nastavení připojení, přihlášení apod.

**ViewModel** je tzv. spojkou mezi Modelem a View. Obaluje data do požadovaného tvaru, zde jsou také funkční metody pro jednotlivé operace s těmito daty. Veškerá logika aplikace by měla být zde.

**View** poskytuje UI psané v XAMLu, je zde také tzv. kód na pozadí v jazyce C#, ve kterém se řeší hlavně obsluha událostí. I zde se však hojně využívá ViewModelu. Jak již bylo naznačeno výše, díky Bindingu je zajištěna okamžitá aktualizace změněných dat, a to jak na straně UI, tak na straně datové struktury. To znamená, že pokud uživatel změní data v nějakém prvku, například do formulářového pole (TextBox) vepíše text, tento se automaticky změní v proměnné, přiřazené přes *Binding*. Stejně to funguje i v opačném případě, tedy když se data změní v proměnné svázané s UI prvkem, uživatel to uvidí.

## 4.2. LINQ

Neboli Language Integrated Query přichází na svět společně s .NET Frameworkem verze 3.5 a přináší podporu dotazování směrem na databázi přímo do programovacího jazyka. Tato integrace je úplná a případné chyby v dotazování se tak projeví již v čase kompilace, nikoli až za běhu programu.

LINQ je možno využít pro práci nad prakticky jakoukoli strukturou dat, ať už se jedná o data relační, hierarchická či jiná. Navíc LINQ nenabízí jen pouhé dotazování, ale umožňuje také snadnou modifikaci dat, čímž se z něj stává velmi mocný nástroj.

LINQ má několik oficiálních, nejpoužívanějších implementací již zabudovaných v sobě:

- *LINQ to Objects* – implementace pro práci se standardními kolekcemi
- *LINQ to SQL* – implementace pro MS SQL Server 2000 a vyšší
- *LINQ to XML* – implementace pro práci nad XML daty
- *LINQ to DataSet* – implementace pro práci s ADO.NET datasety

Aby bylo možno aplikovat LINQ metody dotazování nad konkrétní kolekci, musí tato implementovat rozhraní *IEnumerable<T>*, které je v .NET Frameworku přítomno již od verze 2. Aby však byla zajištěna plná podpora LINQ dotazování, musely být všechny typy implementující ono rozhraní rozšířeny o bohatou škálu různých metod právě pro práci s dotazy nad daty. Tato množina je označována jako *Standart query operators* a obsahuje například metodu *Where* pro omezení výběru z kolekce, metodu *Select* pro vybrání jen určitých dat z kolekce či naformátování dat do požadovaného tvaru a v neposlední řadě také metodu *OrderBy*, která nám umožní data seřadit.

Při vymýšlení LINQ se myslelo také na „staré“ kolekce, neimplementující rozhraní *IEnumerable<T>*. Byla vytvořena zvláštní extension metoda *Cast*, která ze „staré“ kolekce udělá kolekci, nad kterou se již můžeme bez potíží dotazovat. Jen je nutno dodržet podmínku stejného typového obsahu uvnitř převáděné kolekce.

## 5. Závěr

Odbornou praxi jsem nastoupil s nadšením a očekáváním. Těšil jsem se na zapojení se do běžného provozu takto velké společnosti a zároveň jsem se trochu obával, zda-li jsou mé schopnosti dostačující.

Z každého zaměstnance, se kterým jsem se setkal, byla cítit profesionalita a každý byl maximálně ochotný pomoci, když se vyskytl nějaký problém. Práce je mezi zaměstnanci logicky dělena, každý tedy zná své kompetence a své meze.

Mně zadávané úkoly byly v rámci mých schopností, ačkoli jsem se musel přiučit hned několika novým věcem. V rámci společnosti také platí určité metodiky, které jsem si musel nastudovat a jim jsem také musel přizpůsobit styl psaní kódu. Každý úkol pro mne byl něčím novým a něčím zajímavým, každý mne naučil něco nového. Mé řešení úkolů bylo většinou bráno spokojeně bez připomínek, ačkoli mi jich pár bylo vráceno, abych opravil chyby. Protože však tyto chyby byly většinou malého rozsahu, počítám svou práci jako úspěšně splněnou.

Celou praxi hodnotím velmi kladně. Díky této příležitosti jsem se nejen mnohému přiučil, ale také jsem mohl na vlastní kůži zažít pracovní cyklus. Sledovat běh tak velké společnosti, jakou AutoCont CZ a.s. určitě je, bylo místy až fascinující. Jsem hrdý na fakt, že jsem byl přijat do složitějšího soukolí společnosti a věřím, že jsem v průběhu své praxe nezpůsobil žádný zádrhel.

## 6. Použitá literatura

*Video streaming* | SYMBIO [online]. c2011 [cit. 2011-05-16]. Symbio - internetová agentura. Dostupné z WWW: <http://www.symbio.cz/slovník/video-streaming.html>

DAJBÝCH, Václav. *MVVM: Model-View-ViewModel* [online]. 21. dubna 2009 [cit. 2011-05-16]. MVVM: Model-View-ViewModel. Dostupné z WWW: <http://dajbych.net/model-view-viewmodel>

*Struktura společnosti - AutoCont CZ a.s.* [online]. c2010 [cit. 2011-05-16]. Struktura společnosti. Dostupné z WWW: <http://www.autocont.cz/profil-struktura.cml>

*Adaptic - Co je WYSIWYG* [online]. c2011 [cit. 2011-05-16]. WYSIWYG. Dostupné z WWW: <http://www.adaptic.cz/znalosti/slovnicek/wysiwyg/>

*Domovská stránka webu MSDN* [online]. c2011 [cit. 2011-05-16]. Dostupné z WWW: <http://msdn.microsoft.com/cs-cz/>

*Standard ECMA-334 : C# Language Specification* [online]. June 2006 [cit. 2011-05-16]. Standard ECMA-334. Dostupné z WWW: <http://www.ecma-international.org/publications/standards/Ecma-334.htm>

*Javascript - úvod* [online]. 02. května 2011 [cit. 2011-05-16]. Úvod do JavaScriptu. Dostupné z WWW: <http://www.jakpsatweb.cz/javascript/javascript-uvod.html>

*Úvod do LINQ* | *Vyvojar.cz* [online]. 24. ledna 2008 [cit. 2011-05-16]. Úvod do LINQ. Dostupné z WWW: <http://www.vyvojar.cz/Articles/563-uvod-do-linq.aspx>

HOTEK, Mike. *Microsoft SQL Server 2008 : Step by Step*. Redmond (Washington) : Microsoft Press, 2009. 485 s. ISBN 0-7356-2604-9, ISBN-13 978-0-7356-2604-1

ESPOSITO, Dino. *Programming Microsoft ASP.NET 3.5*. Redmont (Washington) : Microsoft Press, 2008. 1126 s. ISBN 0-7356-2527-1, ISBN-13 978-0-7356-2527-3

ARCHER, Tom. *Inside C#*. Redmond (Washington) : Microsoft Press, 2001. 403 s. ISBN 0-7356-1288-9

*Wikipedie* [online]. 24. 5. 2011 [cit. 2011-06-27]. Wikipedie. Dostupné z WWW: [http://cs.wikipedia.org/wiki/C\\_Sharp](http://cs.wikipedia.org/wiki/C_Sharp)